

Package: shinyresults (via r-universe)

September 7, 2024

Type Package

Title A collection of shiny apps and modules to visualize/handle model results

Version 0.31.2

Date 2024-07-09

Description A collection of tools which allow to manipulate and analyze code.

License BSD_2_clause + file LICENSE

URL <https://github.com/pik-piam/shinyresults>,
<https://doi.org/10.5281/zenodo.1478922>

BugReports <https://github.com/pik-piam/shinyresults/issues>

Depends shiny

Imports curl, data.table, doSNOW, foreach, ggplot2, magclass, mip, parallel, plotly, reshape2, snow, traffilight, withr

Suggests covr, lrcode2, quitte (>= 0.3072)

Encoding UTF-8

LazyData no

RoxygenNote 7.3.1

Repository <https://pik-piam.r-universe.dev>

RemoteUrl <https://github.com/pik-piam/shinyresults>

RemoteRef HEAD

RemoteSha b9a9784b82831bc7d7cff54f267f973793a54dea

Contents

shinyresults-package	2
appModelstats	3
appResults	3
appResultsLocal	4

extractVariableGroups	5
modAreaPlot	5
modAreaPlotUI	6
modFilter	6
modFilterUI	8
modLinePlot	8
modLinePlotUI	9
modRunSelect	9
modRunSelectUI	10

Index	11
--------------	-----------

shinyresults-package *shinyresults: A collection of shiny apps and modules to visualize/handle model results*

Description

A collection of tools which allow to manipulate and analyze code.

Author(s)

Maintainer: Jan Philipp Dietrich <dietrich@pik-potsdam.de>

Authors:

- Florian Humpenoeder <humpenoeder@pik-potsdam.de>
- Pascal Sauer <pascal.sauer@pik-potsdam.de>

See Also

Useful links:

- <https://github.com/pik-piam/shinyresults>
- [doi:10.5281/zenodo.1478922](https://doi.org/10.5281/zenodo.1478922)
- Report bugs at <https://github.com/pik-piam/shinyresults/issues>

appModelstats	<i>Analyze Model Statistics</i>
---------------	---------------------------------

Description

Shiny app to analyze statistics collected with [runstatistics](#) and merged with [mergestatistics](#)

Usage

```
appModelstats(
  files = c("https://www.pik-potsdam.de/rd3mod/magpie.rds",
            "https://www.pik-potsdam.de/rd3mod/remind.rds"),
  resultsfolder = NULL
)
```

Arguments

files	path to rds-files from which statistics should be read
resultsfolder	path to a folder containing model results of the corresponding runs

Author(s)

Jan Philipp Dietrich

appResults	<i>appResults</i>
------------	-------------------

Description

appResults allows to explore and visualize time series of modelling results.

Usage

```
appResults(cfg = getOption("appResults"), readFilePar = FALSE, ...)
```

Arguments

cfg	<p>config-file containing information about one or more models on the items listed below. Usually you provide these settings in your .Rprofile as a list constructed as follows (to explore REMIND results please replace all occurrences of magpie with remind):</p> <pre>url <- https://rse.pik-potsdam.de/data/magpie/results/rev1 options(appResults = list(MAgPIE = list(file = paste0(url, "/reduced_overview.rds"), resultsfolder=url, valfile=paste0(url, "/validation.rds"), username="xxx", password="yyy")))</pre> <p>file - Overview file in rds format containing a list of all runs available. To get access to all available filters use overview.rds (takes 15s longer to load) instead</p>
-----	--

of `reduced_overview.rds`. `resultsfolder` - folder in which model results are stored in rds format. `valfile` - validation data. Can be a CSV/MIF file or rds file with a `quite` object (saved with `saveRDS`). `NULL` by default; in this case the user can upload files directly in the tool. `username` - username to access "file" and "resultsfolder". `password` - password to access "file" and "resultsfolder".

`readFilePar` read report data files in parallel (faster) (`TRUE`) or in sequence (`FALSE`)

... additional information to overwrite one of the settings from the `cfg` directly: `file`, `resultsfolder`, `valfile`, `username` or `password`.

Author(s)

Florian Humpenoeder, Jan Philipp Dietrich, Lavinia Baumstark, Pascal Sauer

`appResultsLocal` *appResultsLocal*

Description

version of `appResults` which is optimized to run on a local model folder. In contrast to `appResults`, `appResultsLocal` only requires the path to an output folder (with subfolders for each run).

Usage

```
appResultsLocal(folder = "output/", valfile = NULL)
```

Arguments

`folder` output folder containing the runs to be analyzed as subfolders (e.g. folder "output" in a `MAGPIE` model folder)

`valfile` Path to a validation file, preferably in rds format, but can also be provided as mif (in the latter case it will be converted to rds first). If not path is given the function will look automatically for an validation file in the output folder

Author(s)

Jan Philipp Dietrich, Lavinia Baumstark

See Also

[appResults](#)

extractVariableGroups *extractVariableGroups*

Description

Groups variable names by groups based on the | separators given in the variable names

Usage

```
extractVariableGroups(x)
```

Arguments

x a vector of variable names

Value

a data frame with variables and corresponding groups as columns.

Author(s)

Jan Philipp Dietrich

See Also

[plotstyle.add](#)

Examples

```
x <- c("a|1|aa", "a|2|abc", "a|1|bb", "a|1|cc", "a|3|aa", "a|3|bb", "b|2")
shinyresults::extractVariableGroups(x)
```

modAreaPlot

modAreaPlot Module

Description

Shiny module which works together with [modAreaPlotUI](#) to produce an area plot tab

Usage

```
modAreaPlot(input, output, session, report)
```

Arguments

input, output, session

Default input, output and session objects coming from shiny

report

A reactive containing the report to be visualized

Author(s)

Jan Philipp Dietrich, Florian Humpenoeder

See Also

[modAreaPlotUI](#), [appResults](#)

modAreaPlotUI	<i>modAreaPlot Module</i>
---------------	---------------------------

Description

Shiny module which works together with [modAreaPlot](#) to produce an area plot tab

Usage

```
modAreaPlotUI(id)
```

Arguments

id	id of the filter
----	------------------

Author(s)

Florian Humpenoeder, Jan Philipp Dietrich

See Also

[modAreaPlot](#), [appResults](#)

modFilter	<i>modFilter Module</i>
-----------	-------------------------

Description

Shiny module which works together with [modFilterUI](#) to filter a data set based on user input

Usage

```

modFilter(
  input,
  output,
  session,
  data,
  exclude = NULL,
  showAll = FALSE,
  multiple = NULL,
  xdata = NULL,
  xdataExclude = NULL,
  order = NULL,
  name = NULL,
  preselectYear = NULL,
  preselectMinDate = NULL
)

```

Arguments

input, output, session	Default input, output and session objects coming from shiny
data	A reactive returning a data.table with observations in rows and filter options in columns
exclude	names of columns that should be not used as filter
showAll	FALSE If set to TRUE all available filter are shown and the filter selector is hidden
multiple	vector with booleans for each filter defining whether multiple selections are allowed or not. If information is not provided it is assumed that multiple selection is allowed
xdata	additional data.tables which should be filtered by the same rules as data. If provided the format of the return value changes
xdataExclude	similar to exclude a vector of filters that should be ignored for xdata. Useful if xdata should only filtered for a subset of filters applied to data
order	order the filter should be listed (provided as a vector of filter names). Filter not listed here will be shown after the ones mentioned.
name	name used to identify the filter in the log
preselectYear	if provided the year filter will be preselected with this value
preselectMinDate	if provided the date filter will be preselected with this as lower value

Value

a reactive list with x as the filtered data and xdata containing the list of additional, filtered data element.

Author(s)

Jan Philipp Dietrich

See Also

[modFilterUI](#), [appModelstats](#)

modFilterUI

modFilterUI Module

Description

Corresponding user interface to [modFilter](#) to filter a data set based on user input

Usage

```
modFilterUI(id)
```

Arguments

id id of the filter

Author(s)

Jan Philipp Dietrich

See Also

[modFilter](#), [appModelstats](#)

modLinePlot

modLinePlot Module

Description

Shiny module which works together with [modLinePlotUI](#) to produce a line plot tab

Usage

```
modLinePlot(input, output, session, report, validation)
```

Arguments

input, output, session Default input, output and session objects coming from shiny
report A reactive containing the report to be visualized
validation A reactive containing validation data to be shown

Author(s)

Florian Humpenoeder, Jan Philipp Dietrich

See Also

[modLinePlotUI](#), [appResults](#)

modLinePlotUI	<i>modLinePlot Module</i>
---------------	---------------------------

Description

Shiny module which works together with [modLinePlot](#) to produce a line plot tab

Usage

```
modLinePlotUI(id)
```

Arguments

id	id of the filter
----	------------------

Author(s)

Florian Humpenoeder, Jan Philipp Dietrich

See Also

[modLinePlot](#), [appResults](#)

modRunSelect	<i>modRunSelect Module</i>
--------------	----------------------------

Description

Corresponding server logic to [modRunSelectUI](#) to select modules runs for further analysis

Usage

```
modRunSelect(  
  input,  
  output,  
  session,  
  file,  
  resultsfolder,  
  username = NULL,  
  password = NULL,  
  readFilePar = FALSE  
)
```

Arguments

input, output, session	Default input, output and session objects coming from shiny
file	report data. Can be a CSV/MIF file or rds file with a quitte object (saved with saveRDS). file can also be a vector of rds files. NULL by default; in this case the user can upload files directly in the tool
resultsfolder	folder in which MAGPIE run results are stored. File must come with a overview list called "files"
username	username to be used to access file and resultsfolder
password	password to access file and resultsfolder
readFilePar	read report data files in parallel (faster) (TRUE) or in sequence (FALSE)

Value

a reactive containing a merged data.frame containing results of selected runs

Author(s)

Jan Philipp Dietrich

See Also

[modFilterUI](#), [appModelstats](#)

modRunSelectUI

modRunSelectUI Module

Description

Corresponding user interface to [modRunSelect](#) to select modules runs for further analysis

Usage

```
modRunSelectUI(id)
```

Arguments

id	id of the filter
----	------------------

Author(s)

Jan Philipp Dietrich

See Also

[modFilter](#), [appModelstats](#)

Index

`appModelstats`, [3](#), [8](#), [10](#)
`appResults`, [3](#), [4](#), [6](#), [9](#)
`appResultsLocal`, [4](#)

`extractVariableGroups`, [5](#)

`mergestatistics`, [3](#)
`modAreaPlot`, [5](#), [6](#)
`modAreaPlotUI`, [5](#), [6](#), [6](#)
`modFilter`, [6](#), [8](#), [10](#)
`modFilterUI`, [6](#), [8](#), [8](#), [10](#)
`modLinePlot`, [8](#), [9](#)
`modLinePlotUI`, [8](#), [9](#), [9](#)
`modRunSelect`, [9](#), [10](#)
`modRunSelectUI`, [9](#), [10](#)

`plotstyle.add`, [5](#)

`runstatistics`, [3](#)

`shinyresults (shinyresults-package)`, [2](#)
`shinyresults-package`, [2](#)